



UNIVERSITÀ DEGLI STUDI DI SALERNO

DIPARTIMENTO DI INFORMATICA

Corso di Laurea Triennale in Informatica

**Progettazione e implementazione di
un'infrastruttura di rete con sistema di
monitoraggio**

Relatore

Prof. Christiancarmine Esposito

Università degli Studi di Salerno

Candidato

Samuele Sparno

Matr. 0512114302

ANNO ACCADEMICO 2024/2025

**Il seguente progetto di tesi è stato realizzato
durante il tirocinio curriculare presso**



System Management S.p.A.

Tutor Aziendale

Gianluca De Simone

Abstract

Nell'era digitale, dove le infrastrutture IT rappresentano il cuore delle attività aziendali, garantire la stabilità e l'efficienza della rete è una priorità assoluta. Qualsiasi anomalia, che si tratti di un'interruzione di servizio, di un malfunzionamento di un dispositivo o di un sovraccarico delle risorse, può avere conseguenze significative sulle operazioni aziendali e sulla continuità operativa dell'infrastruttura. Per questo motivo, il monitoraggio della rete e dei dispositivi è diventato un elemento essenziale nella gestione dell'IT, consentendo di rilevare tempestivamente eventuali criticità e intervenire in modo proattivo. Il progetto descritto in questa tesi si focalizza sulla progettazione e realizzazione di un'infrastruttura di rete, seguita dall'implementazione di un sistema di monitoraggio in grado di controllare lo stato delle macchine e dei servizi attivi. L'obiettivo è quello di verificare la continuità operativa delle macchine sotto osservazione, rilevare immediatamente eventuali anomalie e fornire dati dettagliati sulle prestazioni. Inoltre, la possibilità di ricevere notifiche automatiche in caso di criticità consente agli amministratori di rete di intervenire rapidamente, riducendo i tempi di inattività e garantendo una gestione più efficiente delle risorse. Questo progetto ha permesso di testare concretamente l'efficacia di un sistema di monitoraggio su un'infrastruttura realizzata ad hoc, valutandone la capacità di rilevare problemi e generare alert.

Indice

Abstract	1
1 Introduzione	7
1.1 Contesto	7
1.2 Obiettivo	8
1.3 Tematiche di Ricerca e Sviluppo	9
1.4 Struttura dei Contenuti	10
2 Stato dell'arte	11
2.1 Monitoraggio e gestione delle reti	11
2.1.1 Vantaggi del monitoraggio	12
2.1.2 Principali Metodologie e strumenti	13
2.2 Sicurezza delle infrastrutture di rete	14
2.2.1 Progettazione e segmentazione	15
2.2.2 Firewall	16
2.3 Tecnologie e strumenti utilizzati nel progetto	17
2.3.1 Nagios Core e Postfix: monitoraggio e gestione delle notifiche .	17
2.3.2 GNS3 e FortiGate: simulazione di rete e gestione della sicurezza	19

3	Progettazione e implementazione	22
3.1	Progettazione della rete	22
3.1.1	Definizione della topologia	22
3.1.2	Scelta dei dispositivi e strumenti impiegati	23
3.2	Rete di laboratorio in GNS3 per prove preliminari	25
3.2.1	Struttura e configurazione della rete simulata	25
3.3	Implementazione su dispositivi fisici	26
3.3.1	Endpoint: configurazione IP e sistema operativo	27
3.3.2	Switch Cisco: trunking e tagging VLAN	28
3.3.3	Firewall FortiGate: policy di sicurezza e DHCP	29
3.3.4	Router Cisco: gestione del routing	32
3.3.5	Verifica della connettività	32
3.4	Configurazione dei servizi di rete	34
3.4.1	Server HTTP	35
3.4.2	Server FTP	36
3.5	Sistema di monitoraggio	37
3.5.1	Installazione di Nagios Core	37
3.5.2	Configurazione degli host con NSClient++	39
3.6	Sistema di notifiche e alert automatici	42
3.6.1	Installazione e configurazione di Postfix	42
3.6.2	Test e gestione degli alert via email	44
4	Conclusioni	47

Bibliografia

48

Elenco delle figure

2.1	Le funzioni di un sistema di monitoraggio della rete [10]	12
2.2	Architettura e comunicazione tra Nagios e NRPE [12]	13
2.3	Segmentazione di una rete [18]	15
2.4	Interfaccia Policy Firewall FortiGate [4]	17
2.5	Flusso di Monitoraggio e Notifica in Nagios Core	19
2.6	Esempio di infrastruttura di rete in ambiente di simulazione GNS3 [8]	20
2.7	Dispositivo firewall FortiGate 200F [5]	21
3.1	Architettura della rete finale progettata	24
3.2	Rete di prova creata su GNS3	26
3.3	Dashboard VMware della macchine configurate come endpoint	27
3.4	Stato delle interfacce dello switch tramite MobaXterm	29
3.5	Dashboard del firewall FortiGate	29
3.6	Configurazione dell'interfaccia vlanServizi	30
3.7	Dashboard finale delle interfacce del firewall	31
3.8	Policy di sicurezza configurate sul firewall	32

3.9	Verifica della connettività Internet da un endpoint tramite comando ping	33
3.10	Verifica della connettività inter-VLAN	33
3.11	Monitoraggio del traffico delle policy firewall tramite la colonna «Bytes»	34
3.12	Pagina di default di IIS	35
3.13	Test di connessione al server FTP da FileZilla Client	36
3.14	Verifica dello stato del servizio Nagios	38
3.15	Homepage dell'interfaccia web di Nagios Core	38
3.16	Interfaccia web di Nagios dello stato degli host	42
3.17	Notifica email per stato DOWN di un host	44
3.18	Notifica email generata per inattività del servizio HTTP sull'host pc2windows	45
3.19	Notifica email generata per superamento della soglia di WARNING del carico CPU	46

Capitolo 1

Introduzione

1.1 Contesto

Nel panorama digitale odierno, il monitoraggio della rete è diventato una componente essenziale per le aziende di tutte le dimensioni. Man mano che la tecnologia continua ad avanzare e la dipendenza dall'infrastruttura digitale aumenta, le organizzazioni devono assicurare elevate prestazioni e massima sicurezza all'interno delle proprie reti. Il monitoraggio della rete consente alle aziende di rilevare e risolvere in modo proattivo i problemi prima che degenerino, riducendo al minimo i tempi di inattività e massimizzando la produttività. Fondamentalmente, il monitoraggio della rete prevede l'osservazione e l'analisi in tempo reale del traffico, dei sistemi e dei dispositivi della rete. Ciò permette ai professionisti IT di acquisire informazioni cruciali sulle prestazioni della rete, sull'efficienza dei dispositivi connessi e sulle potenziali minacce alla sicurezza. Definendo parametri specifici e soglie prestazionali, gli amministratori possono monitorare e misurare in modo efficace le prestazioni della rete, prendendo decisioni informate e adottando misure proattive quando necessario. Valutando regolarmente le prestazioni della rete, le aziende possono identificare eventuali colli di bottiglia o problemi che potrebbero ostacolare la produttività e l'efficienza. Senza una strategia completa di monitoraggio della rete,

le aziende potrebbero correre il rischio di subire guasti di rete costosi e dannosi [7].

1.2 Obiettivo

Questo lavoro di tesi si propone di progettare, implementare e testare un sistema di monitoraggio di rete all'interno di un'infrastruttura appositamente costruita. L'obiettivo principale è sviluppare una soluzione in grado di rilevare tempestivamente eventuali anomalie, come il downtime dei dispositivi e dei servizi attivi (HTTP, FTP), garantendo anche un controllo costante su parametri cruciali come CPU, RAM e spazio disco. Per raggiungere tale obiettivo, il progetto è stato articolato nelle seguenti fasi operative:

1. **Progettazione e configurazione dell'infrastruttura di rete:** simulata inizialmente in GNS3 e poi implementata su dispositivi fisici, con segmentazione tramite VLAN e opportune politiche di sicurezza.
2. **Installazione e configurazione di Nagios Core:** su macchina Ubuntu dedicata, per il monitoraggio di host, servizi e risorse.
3. **Configurazione degli endpoint Windows con NSClient++:** per raccogliere informazioni di sistema e servizi in esecuzione, da inviare a Nagios.
4. **Implementazione degli alert via email con Postfix:** per notificare tempestivamente eventuali anomalie agli amministratori di rete.
5. **Fase di test e valutazione:** simulazione di errori reali per verificare l'efficacia del sistema di monitoraggio e delle notifiche.

1.3 Tematiche di Ricerca e Sviluppo

La realizzazione del progetto ha richiesto un'approfondita analisi di diverse aree tematiche, tra cui la progettazione di rete, la sicurezza informatica e il monitoraggio delle infrastrutture. La configurazione dell'infrastruttura di rete è stata sviluppata sulla base dei principi e delle metodologie illustrate nei manuali *CCNA 200-301 Official Cert Guide, Volume 1 e 2*, con particolare attenzione al subnetting, alla segmentazione tramite VLAN, ai protocolli di comunicazione e alla configurazione dei dispositivi Cisco. In merito alla gestione della sicurezza, il *FortiGate Administrator Study Guide – FortiOS 7.4* ha fornito le linee guida essenziali per la configurazione del firewall, la segmentazione del traffico di rete e l'implementazione delle policy di accesso. L'uso di GNS3 ha consentito di simulare e testare la configurazione della rete in un ambiente virtuale, riducendo il rischio di errori durante l'implementazione su dispositivi fisici. Inoltre, per l'accesso remoto ai dispositivi Cisco è stato utilizzato il software *MobaXterm*, che ha permesso una gestione semplificata tramite interfaccia SSH, facilitando le operazioni di configurazione in ambiente reale. Questa fase preliminare ha permesso di verificare il corretto funzionamento della connettività, la gestione delle VLAN e l'applicazione delle regole firewall, garantendo una transizione più sicura e controllata verso l'infrastruttura reale. Per l'implementazione del sistema di monitoraggio, è stata consultata la documentazione ufficiale di *Nagios Core*, utilizzata per la configurazione del monitoraggio di host e servizi, e quella di *Postfix*, adottata per la gestione delle notifiche automatiche via email.

1.4 Struttura dei Contenuti

L'elaborato è articolato come segue:

- Il **Capitolo 2** analizza lo stato dell'arte in ambito di monitoraggio di rete, sicurezza e i strumenti utilizzati.
- Il **Capitolo 3** descrive nel dettaglio il processo di progettazione e implementazione dell'infrastruttura di rete, sia in ambiente di simulazione che su dispositivi fisici.
- Il **Capitolo 4** presenta i test effettuati e i risultati ottenuti, valutando l'efficacia del sistema di monitoraggio implementato.
- Il **Capitolo 5** contiene le conclusioni del lavoro, con una valutazione complessiva dell'esperienza svolta e dei risultati ottenuti.

Capitolo 2

Stato dell'arte

2.1 Monitoraggio e gestione delle reti

Nel mondo odierno, il termine monitoraggio della rete è ampiamente diffuso nell'industria IT. Il monitoraggio della rete è un processo essenziale dell'IT che consiste nell'osservare e analizzare continuamente tutti i componenti della rete, tra cui router, switch, firewall, server e macchine virtuali (VM), al fine di rilevare guasti, valutare le prestazioni e garantirne la massima disponibilità e ottimizzazione. Uno degli aspetti fondamentali del monitoraggio della rete è la sua natura proattiva. Identificare tempestivamente problemi di prestazioni e colli di bottiglia consente di individuare eventuali anomalie fin dalle prime fasi, riducendo il rischio di interruzioni del servizio. Un monitoraggio proattivo ed efficiente dei server può prevenire downtime di rete o guasti critici, migliorando l'affidabilità e la continuità operativa dell'infrastruttura IT. [10]

2.1.1 Vantaggi del monitoraggio

Il monitoraggio della rete offre numerosi vantaggi essenziali per garantire un'infrastruttura IT efficiente e sicura. La flessibilità consente di personalizzare il monitoraggio in base alle esigenze specifiche, fornendo una panoramica chiara dello stato della rete e permettendo agli amministratori di identificare rapidamente criticità grazie a widget e grafici in tempo reale. L'alta disponibilità assicura la connettività continua, rilevando immediatamente guasti o interruzioni e attivando automaticamente un sistema di failover che garantisce il 100% di uptime, con notifiche tempestive in caso di problemi. La scalabilità permette di adattare il monitoraggio all'espansione della rete aziendale, individuando e aggiungendo nuovi dispositivi in modo automatico, mentre la sicurezza protegge dall'accesso non autorizzato e da potenziali minacce informatiche, limitando i permessi degli utenti e verificando l'integrità dei file di sistema. Infine, la compatibilità multi-vendor garantisce il supporto per dispositivi di diversi produttori, semplificando il monitoraggio di ambienti di rete complessi e ibridi, permettendo un controllo efficace su un'infrastruttura eterogenea. [10]



Figura 2.1: Le funzioni di un sistema di monitoraggio della rete [10]

2.1.2 Principali Metodologie e strumenti

Esistono diverse metodologie per il monitoraggio delle reti, ognuna con caratteristiche specifiche che ne determinano l'efficacia a seconda del contesto di utilizzo. La scelta del metodo più adatto dipende dalle esigenze dell'infrastruttura, dalle risorse disponibili e dal livello di dettaglio richiesto nell'analisi. Le principali metodologie di monitoraggio includono:

Monitoraggio basato su agenti: un sistema basato su agenti prevede l'assegnazione di un agente, un programma di monitoraggio, a ogni dispositivo in rete. In questo modo, ogni agente può accedere all'hardware in modo considerevole, ottenendo molte informazioni dettagliate su ciascuno di essi [6]. Un esempio di agente è NRPE (Nagios Remote Plugin Executor), utilizzato nel sistema di monitoraggio Nagios. Per garantire la sicurezza nella comunicazione tra il server Nagios e l'agente NRPE, viene utilizzato di default il protocollo SSL (Secure Sockets Layer), che permette la crittografia dei dati trasmessi, proteggendoli da intercettazioni e garantendo integrità e riservatezza [12].

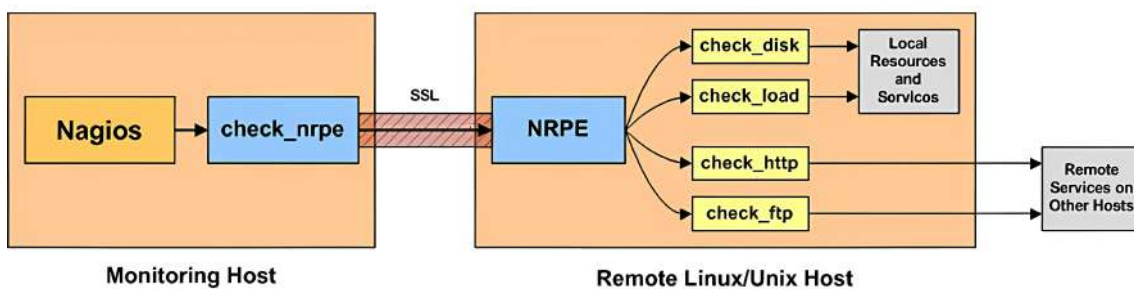


Figura 2.2: Architettura e comunicazione tra Nagios e NRPE [12]

Monitoraggio senza agenti: è generalmente installato on-premise su una workstation o un server connesso fisicamente alla rete e dotato dei privilegi di accesso necessari per monitorare i servizi. Non richiede installazioni su ogni dispositivo della rete e può rilevare automaticamente i dispositivi, ma presenta lo svantaggio di richiedere un sistema dedicato con adeguate risorse di elaborazione; in caso contrario, potrebbe essere necessario acquistare un hardware apposito per supportare il software di monitoraggio [6].

Analisi del traffico: è il processo di esaminare attentamente i dati che viaggiano attraverso una rete informatica, fornendo informazioni preziose sulla salute e sullo stato della rete, nonché sull'attività dei dispositivi e degli utenti che la utilizzano. Ciò è possibile tramite strumenti come Wireshark, ampiamente utilizzato da professionisti della sicurezza, amministratori di rete [17].

2.2 Sicurezza delle infrastrutture di rete

L'infrastruttura di rete è quella parte dell'infrastruttura IT che include l'hardware, il software, i sistemi e i dispositivi che abilitano i flussi dei dati all'interno dell'organizzazione, e che permette di connettere utenti, dispositivi, applicazioni, Internet e altro ancora. Poiché costituisce il punto di collegamento con il mondo esterno, l'infrastruttura di rete è anche un elemento vulnerabile, di cui vanno garantite sicurezza e protezione. Se non si adottano la giusta infrastruttura di rete e le necessarie procedure di sicurezza, si può incorrere in esperienze utente di scarsa qualità e subire attacchi alla rete, con conseguenze sulla produttività e sull'efficienza dei dipendenti. [16]

2.2.1 Progettazione e segmentazione

La progettazione accurata della rete è fondamentale per ottimizzare le prestazioni e garantire la sicurezza. La segmentazione della rete è una politica di sicurezza che prevede la suddivisione della rete aziendale in segmenti più piccoli, isolati tra loro. Ogni segmento funziona come una rete indipendente, offrendo ai team di sicurezza un maggiore controllo sul traffico che attraversa i loro sistemi. Questo approccio ha lo scopo di limitare la diffusione di eventuali minacce e proteggere le risorse aziendali da accessi non autorizzati. Nelle configurazioni di rete, infatti, vengono integrate regole che determinano in che modo utenti e dispositivi all'interno delle sottoreti possono connettersi tra loro. Questa rigida organizzazione della rete rende molto più difficile la propagazione delle minacce e consente di isolare rapidamente eventuali attacchi in modo efficace. [2]

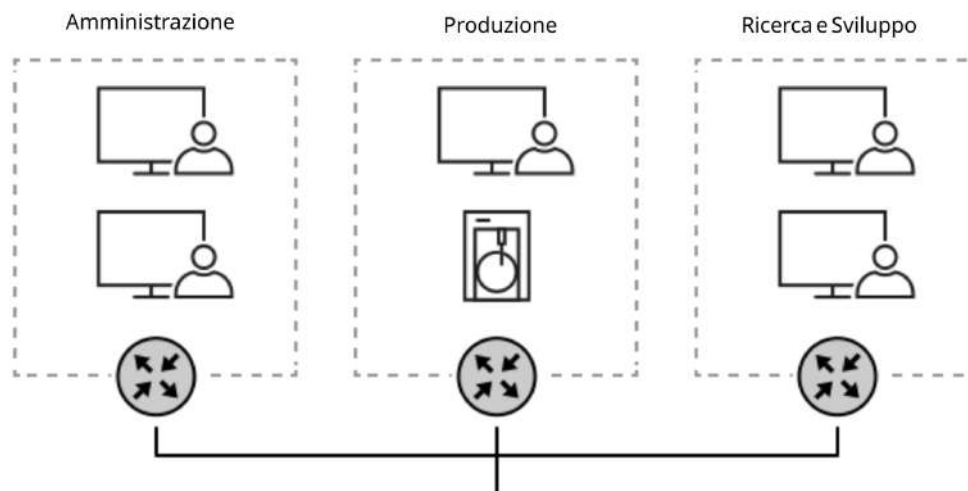


Figura 2.3: Segmentazione di una rete [18]

2.2.2 Firewall

Un firewall è un dispositivo di sicurezza di rete progettato per monitorare, filtrare e controllare il traffico di rete in entrata e in uscita in base a regole di sicurezza predeterminate. Lo scopo principale di un firewall è stabilire una barriera tra una rete interna attendibile e reti esterne non attendibili. I firewall sono disponibili sia in forma hardware che software e funzionano ispezionando i pacchetti di dati e determinando se autorizzarli o bloccarli in base a una serie di regole. [1]

Le regole del firewall possono identificare il traffico consentito o bloccato in base a diverse caratteristiche, tra cui:

- **Indirizzo IP di origine:** identifica la sorgente del traffico. Un'organizzazione può impedire l'accesso da specifici indirizzi IP o intervalli IP. In alternativa, alcuni computer o servizi possono essere resi accessibili solo a determinati indirizzi IP autorizzati. [9]
- **Indirizzo IP di destinazione:** indica la destinazione del traffico. Ad esempio, un'azienda può impedire agli utenti di accedere a siti web noti per essere dannosi o non conformi alle policy aziendali. [9]
- **Tipologia di servizio:** consente di filtrare il traffico in base al tipo di servizio o protocollo utilizzato. Ad esempio, il firewall può essere configurato per consentire solo richieste di ping (ICMP) oppure per bloccare specifici servizi, come il traffico HTTP o FTP non autorizzato.

2.3. Tecnologie e strumenti utilizzati nel progetto

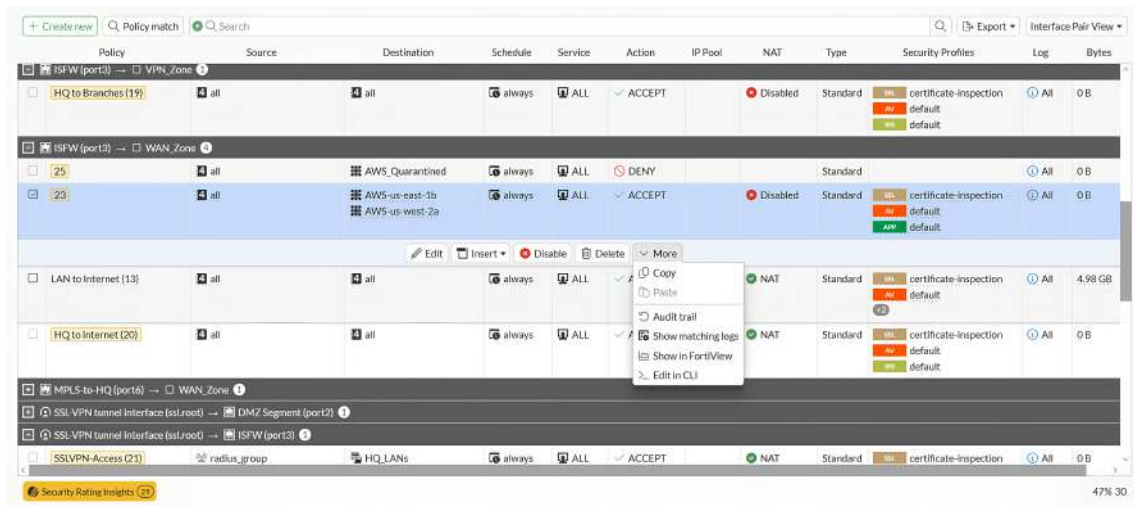


Figura 2.4: Interfaccia Policy Firewall FortiGate [4]

2.3 Tecnologie e strumenti utilizzati nel progetto

Durante il progetto sono stati impiegati diversi strumenti per il monitoraggio della rete, la gestione delle notifiche, la simulazione dell'infrastruttura e il controllo della sicurezza. Nello specifico, Nagios Core e Postfix sono stati utilizzati per il monitoraggio e l'invio di notifiche, mentre GNS3 e FortiGate sono stati adottati per la simulazione della rete e la gestione della sicurezza.

2.3.1 Nagios Core e Postfix: monitoraggio e gestione delle notifiche

Nagios Core è una soluzione open-source dedicata al monitoraggio di sistemi e reti. Il suo scopo principale è quello di controllare host e servizi definiti dall'utente, segnalando tempestivamente eventuali anomalie o ripristini. Sebbene inizialmente sviluppato per ambienti Linux, è compatibile con gran parte dei sistemi Unix. Tra le sue principali funzionalità vi sono il monitoraggio di servizi di rete come SMTP, POP3, HTTP e PING, oltre alla verifica delle risorse degli host, quali il carico del

processore e l'utilizzo dello spazio disco. Nagios Core mette inoltre a disposizione un'interfaccia web che consente agli amministratori di rete di visualizzare in tempo reale lo stato dell'infrastruttura, accedere allo storico delle notifiche, analizzare i log degli eventi e ottenere informazioni cruciali per una corretta gestione operativa.

Un'altra caratteristica rilevante di Nagios Core è il sistema avanzato di notifiche automatiche, che consente agli amministratori di essere prontamente informati su eventuali problemi o recuperi dei servizi monitorati. È possibile personalizzare le modalità di allerta utilizzando diversi canali, quali email, SMS e altri metodi configurabili. [13] Per il monitoraggio delle macchine Windows è stato utilizzato NSClient++, un agente compatibile con Nagios che consente di raccogliere informazioni dettagliate su CPU, RAM, spazio su disco e stato dei servizi. Installato localmente sui client Windows, NSClient++ comunica in modo sicuro con il server Nagios, rispondendo alle richieste di controllo tramite plugin dedicati. In questo contesto, Postfix svolge il ruolo fondamentale di Mail Transfer Agent (MTA), ovvero un software server di posta elettronica specializzato nella gestione dell'invio, ricezione e consegna delle email [15]. Progettato principalmente per sistemi Unix, Postfix assicura uno smistamento affidabile ed efficiente dei messaggi tra i server di posta grazie alla sua architettura modulare basata su code di elaborazione, ottimizzando il trattamento delle email sia in entrata sia in uscita. Una caratteristica chiave di Postfix è la capacità di garantire continuità nella consegna dei messaggi, anche in presenza di interruzioni improvvise: qualora il server dovesse arrestarsi durante un trasferimento, il processo può essere ripreso dall'ultima coda in cui l'email era stata memorizzata. Inoltre, Postfix supporta protocolli standard di comunicazione come TCP/IP, assicurando così massima compatibilità e affidabilità. Grazie alla sua efficienza e scalabilità, Postfix risulta adatto a diversi scenari operativi, dalle piccole reti aziendali fino alle infrastrutture più complesse. [14]

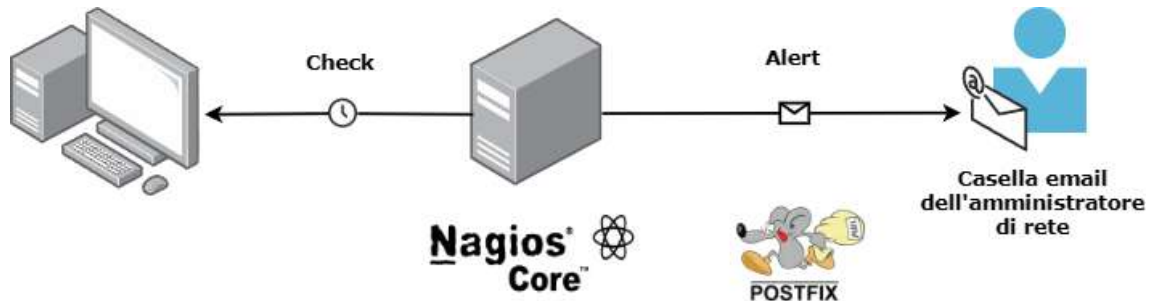


Figura 2.5: Flusso di Monitoraggio e Notifica in Nagios Core

2.3.2 GNS3 e FortiGate: simulazione di rete e gestione della sicurezza

GNS3 (Graphical Network Simulator 3) è un software open-source, gratuito, utilizzato ampiamente per emulare, configurare, testare e risolvere problemi relativi alle reti informatiche. Una delle caratteristiche principali di GNS3 è la sua flessibilità: esso permette infatti di creare e gestire topologie di rete complesse, partendo da poche apparecchiature virtualizzate eseguite localmente su un computer portatile, fino ad arrivare a reti distribuite su server fisici o infrastrutture cloud. L'utilizzo di GNS3 consente agli utenti di testare in tempo reale comportamenti reali delle reti, permettendo simulazioni precise ed affidabili, senza la necessità di possedere costoso hardware fisico dedicato. Questo rende GNS3 uno strumento particolarmente utile non solo per l'apprendimento e la formazione universitaria e professionale, ma anche per lo sviluppo di proof-of-concept, troubleshooting e per la realizzazione di ambienti di laboratorio altamente personalizzati e adattabili alle esigenze specifiche di aziende ed enti di formazione. [3]

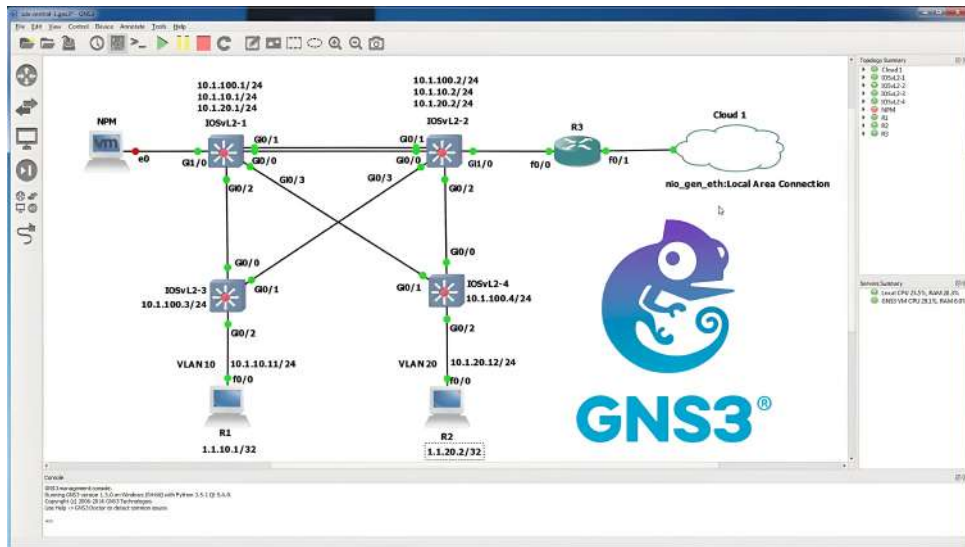


Figura 2.6: Esempio di infrastruttura di rete in ambiente di simulazione GNS3 [8]

Il firewall FortiGate, sviluppato da FortiNet, rappresenta una soluzione innovativa ed è attualmente uno dei leader nel campo dei firewall, in grado di garantire elevati standard di sicurezza e protezione delle reti aziendali. FortiGate dispone di una doppia modalità di configurazione: un'interfaccia web intuitiva, facilmente utilizzabile anche dagli utenti meno esperti, che consente una rapida gestione e visualizzazione delle impostazioni di sicurezza, e una modalità tramite terminale, rivolta principalmente ad amministratori esperti che preferiscono utilizzare comandi diretti per una gestione più precisa e dettagliata del sistema.

La sua versatilità permette un'efficace integrazione sia in ambienti cloud, sia in infrastrutture ibride e fisiche, adattandosi così alle più svariate esigenze operative delle aziende. Una caratteristica distintiva di FortiGate è la sicurezza end-to-end garantita da aggiornamenti continui in tempo reale, grazie alla presenza di una specifica unità hardware dedicata (SPU - Security Processing Unit) che migliora significativamente l'esperienza utente e le prestazioni del sistema. Tra le principali funzionalità offerte, FortiGate include un avanzato sistema di filtraggio del traffico web, che analizza milioni di URL per bloccare in tempo reale contenuti malevoli o sconosciuti. Fondamentale è anche il filtraggio DNS, che offre visibilità totale e

2.3. Tecnologie e strumenti utilizzati nel progetto

protezione completa da tecniche come tunneling e infiltrazioni, insieme ad avanzati meccanismi anti-botnet e reputazione IP per neutralizzare attacchi web diretti. [11]



Figura 2.7: Dispositivo firewall FortiGate 200F [5]

Capitolo 3

Progettazione e implementazione

3.1 Progettazione della rete

La progettazione dell'infrastruttura è stata la fase iniziale del progetto e ha avuto lo scopo di definire una rete stabile, sicura e adatta al successivo monitoraggio. Questa attività ha richiesto una valutazione dei requisiti aziendali e delle tecnologie disponibili, con particolare attenzione all'organizzazione logica del traffico, alla scalabilità della rete e all'integrazione di strumenti di controllo. Nei paragrafi seguenti vengono analizzati gli aspetti principali che hanno guidato le scelte progettuali: dalla definizione della topologia alla selezione dei dispositivi.

3.1.1 Definizione della topologia

La topologia progettata è di tipo gerarchico ed è basata sulla segmentazione logica della rete attraverso VLAN. Questa suddivisione permette di isolare i diversi ambiti funzionali (applicativi e server di monitoraggio) e di applicare policy di sicurezza più efficaci. Il disegno della rete è stato pensato per semplificare l'instradamento del traffico, garantire la scalabilità futura e facilitare l'integrazione con il sistema di monitoraggio previsto.

3.1.2 Scelta dei dispositivi e strumenti impiegati

Per la realizzazione della rete fisica sono stati selezionati dispositivi di fascia enterprise, scelti per la loro affidabilità e compatibilità con ambienti di produzione complessi.

In particolare, è stato impiegato un *virtualizzatore VMware* per la creazione degli endpoint virtualizzati. Per la segmentazione della rete tramite VLAN è stato utilizzato uno *switch Cisco*, configurato con porte in modalità trunk e gestione del traffico inter-VLAN mediante il protocollo 802.1Q. Il *router Cisco* ha permesso il collegamento verso reti esterne tramite connessione cellulare (SIM), garantendo l'accesso a Internet. Il *firewall FortiGate* è stato configurato per la creazione e la gestione delle sub-interfacce associate alle VLAN, per l'erogazione del servizio DHCP e per l'applicazione di policy di sicurezza personalizzate per ciascun segmento di rete. Infine, gli strumenti *Nagios Core*, *NSClient++* e *Postfix* sono stati adottati per il monitoraggio centralizzato delle risorse e per l'invio automatizzato di notifiche in caso di anomalie.

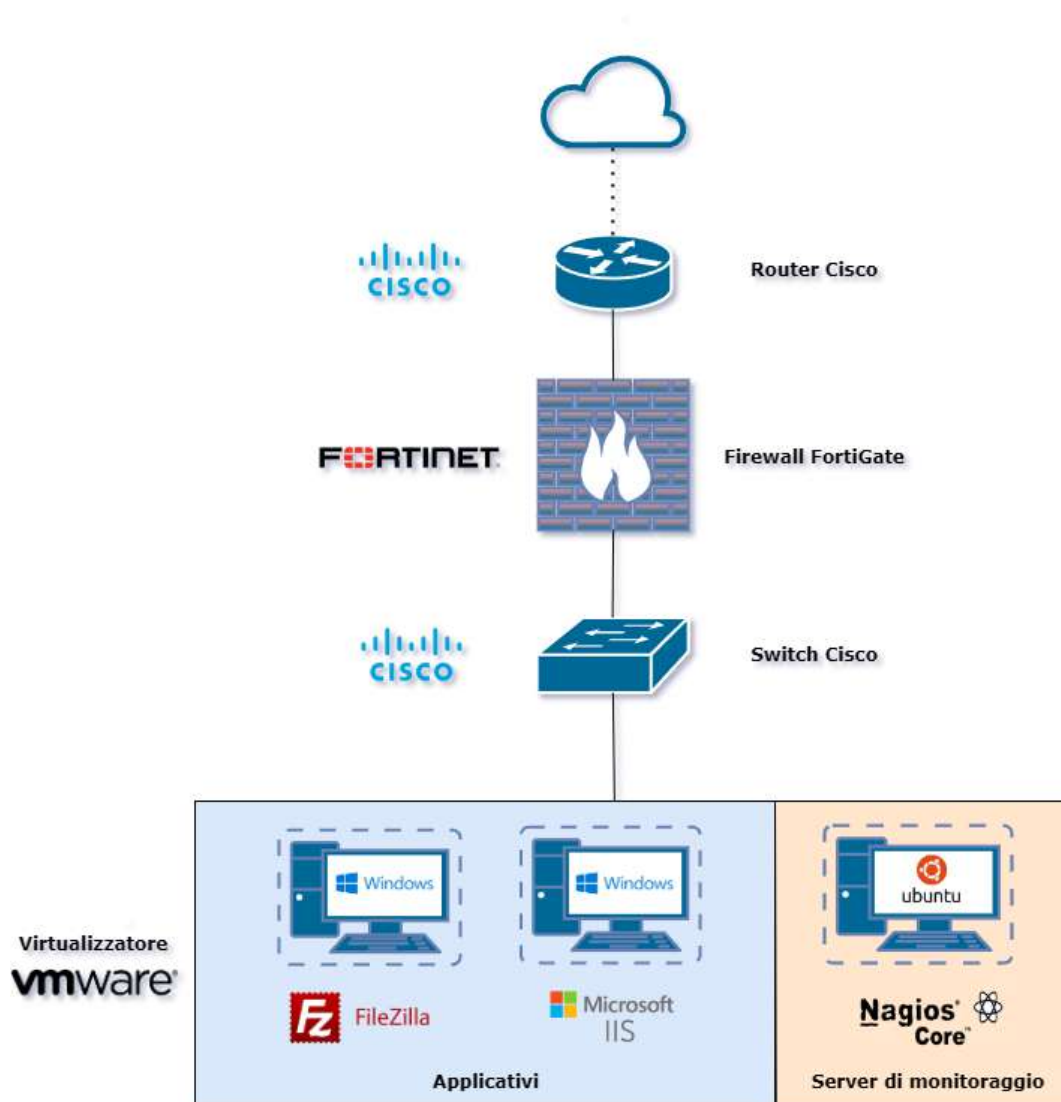


Figura 3.1: Architettura della rete finale progettata

3.2 Rete di laboratorio in GNS3 per prove preliminari

Nel corso del progetto è stata realizzata una rete di laboratorio all'interno dell'ambiente di simulazione GNS3, con l'obiettivo di sperimentare e validare le configurazioni che sarebbero poi state applicate alla rete fisica finale. Questa fase preliminare ha avuto un ruolo fondamentale nel ridurre il rischio di errori, permettendo di verificare in anticipo la correttezza delle impostazioni relative a VLAN, routing, assegnazione IP e firewall policy. La rete simulata è pensata appositamente per eseguire prove rapide ed efficaci in un ambiente controllato attraverso l'utilizzo di GNS3. La simulazione ha inoltre permesso di prendere confidenza con le interfacce di gestione dei dispositivi e con i comandi utilizzati, facilitando la successiva implementazione nel contesto reale.

3.2.1 Struttura e configurazione della rete simulata

La rete di test è stata composta da dispositivi virtuali configurati per replicare gli scenari più rilevanti del progetto finale. Sono stati inseriti host, ciascuno assegnato a VLAN distinte, con indirizzamenti IP specifici per ogni segmento. Gli switch sono stati configurati in modalità trunk, con supporto al protocollo IEEE 802.1Q per il tagging VLAN. Il firewall FortiGate, integrato nella rete simulata, è stato configurato per testare la creazione delle policy di sicurezza tra le VLAN e il funzionamento del servizio DHCP. La gestione è avvenuta tramite l'interfaccia web WebTerm, utilizzata anche per il monitoraggio del traffico e il debug delle configurazioni. Infine, sono stati effettuati test di connettività tra gli host tramite strumenti diagnostici come ping e traceroute, verificando la corretta propagazione del traffico tra le sottoreti, la funzionalità delle regole firewall e la corretta assegnazione degli indirizzi IP.

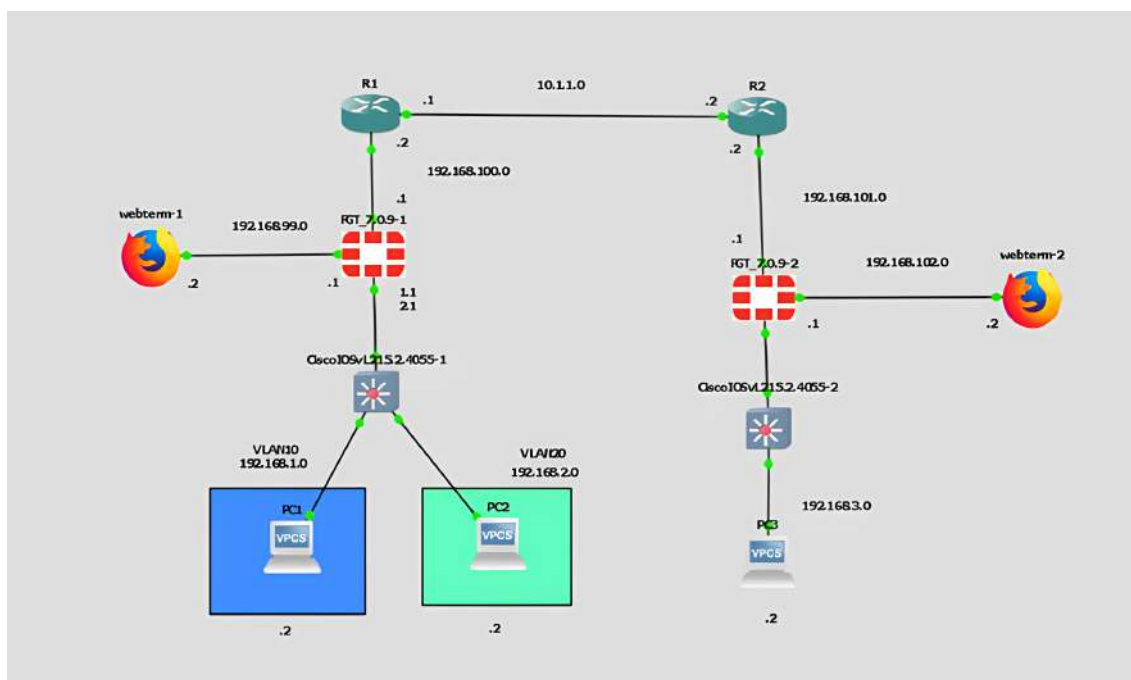


Figura 3.2: Rete di prova creata su GNS3

3.3 Implementazione su dispositivi fisici

Dopo aver definito l'architettura logica della rete, si è passati alla fase di implementazione fisica dell'infrastruttura, grazie alla disponibilità di dispositivi reali forniti dall'azienda ospitante nell'ambito del tirocinio curriculare. A differenza della rete simulata in GNS3, che ha avuto un ruolo esclusivamente esplorativo e formativo, la rete fisica realizzata corrisponde fedelmente all'architettura logica delineata nella fase di progettazione (paragrafo 3.1). Il cablaggio manuale ha rappresentato un elemento fondamentale dell'implementazione, richiedendo particolare cura nella disposizione e nel collegamento fisico dei dispositivi. La corretta assegnazione delle porte, la coerenza nella mappatura delle VLAN e la compatibilità tra gli apparati sono stati aspetti centrali al fine di assicurare un corretto funzionamento e avere una chiara disposizione dei collegamenti, funzionale alle successive configurazioni dell'infrastruttura.

3.3.1 Endpoint: configurazione IP e sistema operativo

Sono stati utilizzati due tipi di endpoint: una macchina Ubuntu, configurata con un indirizzo IP statico, e due macchine Windows, configurate per ottenere dinamicamente un IP tramite il server DHCP del firewall. L'assegnazione dell'IP statico alla macchina Ubuntu è risultata fondamentale per garantirne la raggiungibilità costante da parte degli altri dispositivi di rete, condizione necessaria per i successivi servizi e attività di monitoraggio. L'indirizzo fisso, infatti, consente di evitare problemi legati a variazioni dinamiche e rende stabile la configurazione delle comunicazioni verso tale host. Al contrario, per i client Windows, che non ricoprono ruoli critici all'interno della rete, è stata scelta una configurazione dinamica tramite DHCP, che semplifica la gestione degli indirizzi IP.

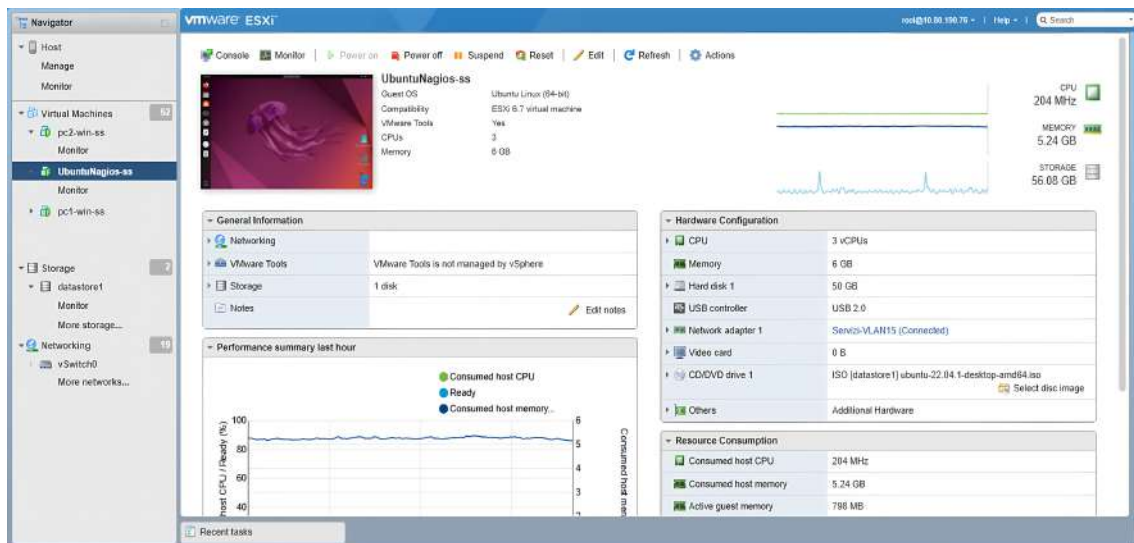


Figura 3.3: Dashboard VMware della macchine configurate come endpoint

3.3.2 Switch Cisco: trunking e tagging VLAN

L'accesso allo switch Cisco è stato effettuato da remoto tramite il software MobaX-term, che ha consentito di stabilire una sessione SSH per l'interazione con l'interfaccia a riga di comando del dispositivo. Dal punto di vista fisico, lo switch è stato collegato al virtualizzatore VMware tramite un'unica porta, a cui sono connessi tutti gli endpoint virtuali, e al firewall FortiGate tramite una seconda porta trunk. Di conseguenza, la configurazione ha riguardato due collegamenti principali: uno verso il virtualizzatore e uno verso il firewall.

Lo switch è stato configurato per:

- Abilitare il trunking sulla porta di collegamento con il firewall, utilizzando il protocollo IEEE 802.1Q per il tagging del traffico VLAN;
- Creare le VLAN necessarie alla segmentazione logica della rete secondo la progettazione definita, in particolare la VLAN 15, riservata alle macchine che forniscono servizi come il monitoraggio, e la VLAN 16, destinata agli endpoint applicativi da monitorare;
- Impostare la porta verso il virtualizzatore VMware in modalità trunk per permettere il transito delle VLAN definite.

3.3. Implementazione su dispositivi fisici

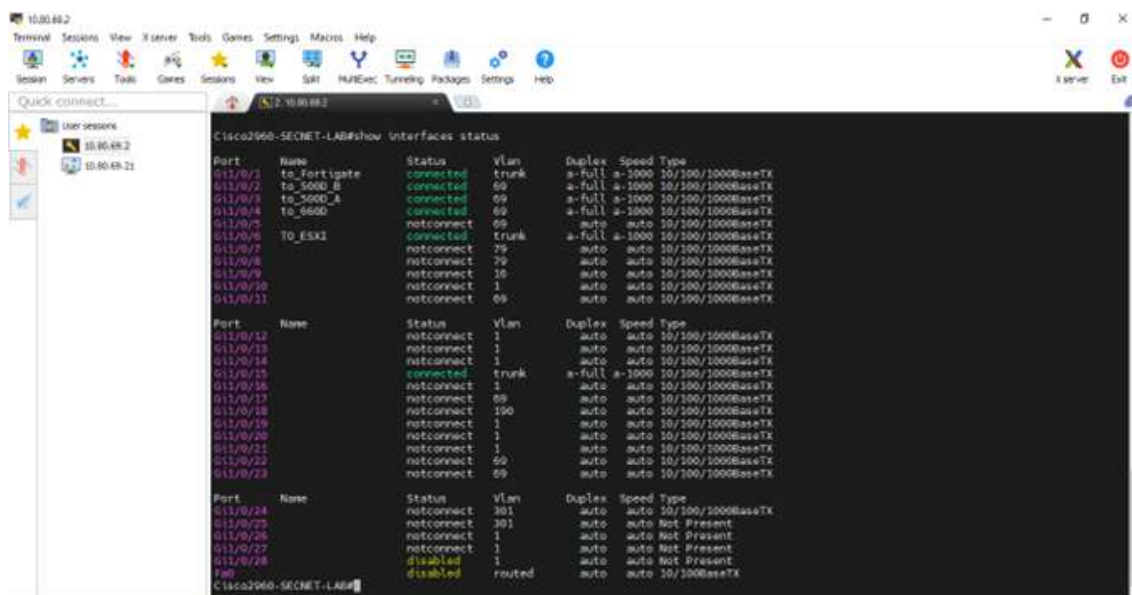


Figura 3.4: Stato delle interfacce dello switch tramite MobaXterm

3.3.3 Firewall FortiGate: policy di sicurezza e DHCP

Per la configurazione del firewall è stata utilizzata l'interfaccia grafica proprietaria dei dispositivi FortiGate. Questa scelta ha permesso di semplificare le operazioni di configurazione, evitando l'utilizzo della linea di comando e rendendo il processo più immediato e accessibile.

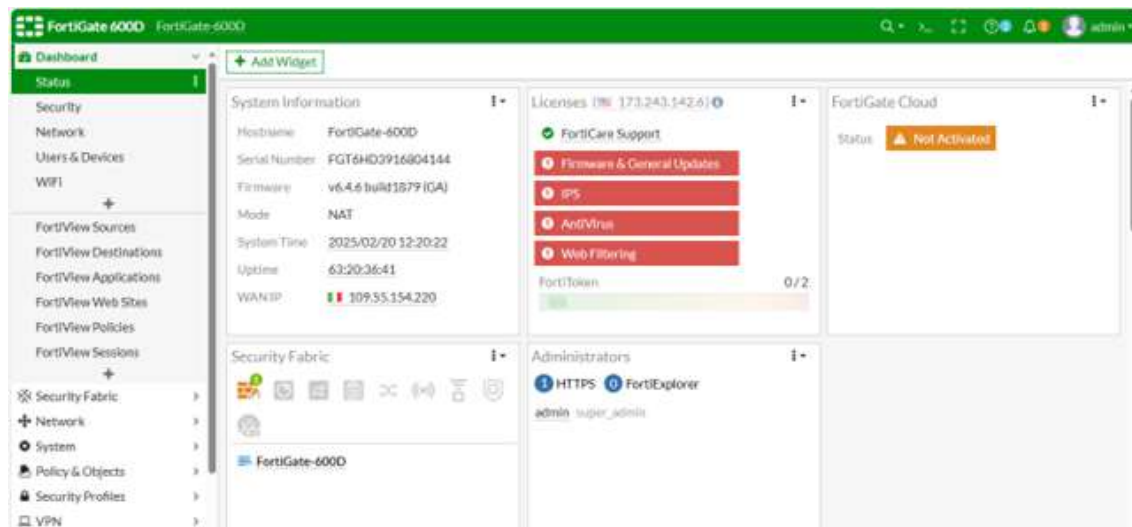


Figura 3.5: Dashboard del firewall FortiGate

3.3. Implementazione su dispositivi fisici

Come primo passo, sono state configurate le interfacce del firewall necessarie per il collegamento con gli altri dispositivi di rete: la porta 9 per il collegamento al router e la porta 15 per il collegamento allo switch. Per la porta 9 è stata creata un'interfaccia fisica, assegnandole un indirizzo IP statico e la relativa subnet mask. Per la porta 15, invece, si è proceduto alla creazione di sub-interfacce VLAN, utilizzando il protocollo 802.1Q per l'incapsulamento dei tag VLAN. In questo modo, la porta agisce come trunk, consentendo il transito del traffico di più VLAN attraverso un'unica interfaccia fisica. A ciascuna sub-interfaccia è stato assegnato un IP di gateway per la rispettiva VLAN, permettendo al firewall di gestire e filtrare il traffico tra le diverse reti segmentate.

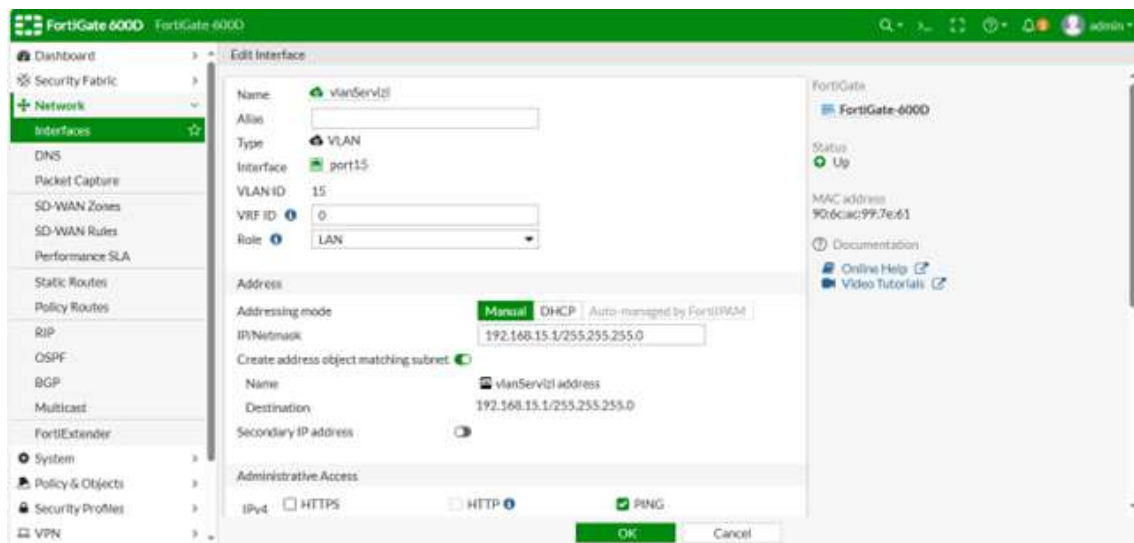


Figura 3.6: Configurazione dell'interfaccia vlanServizi

3.3. Implementazione su dispositivi fisici

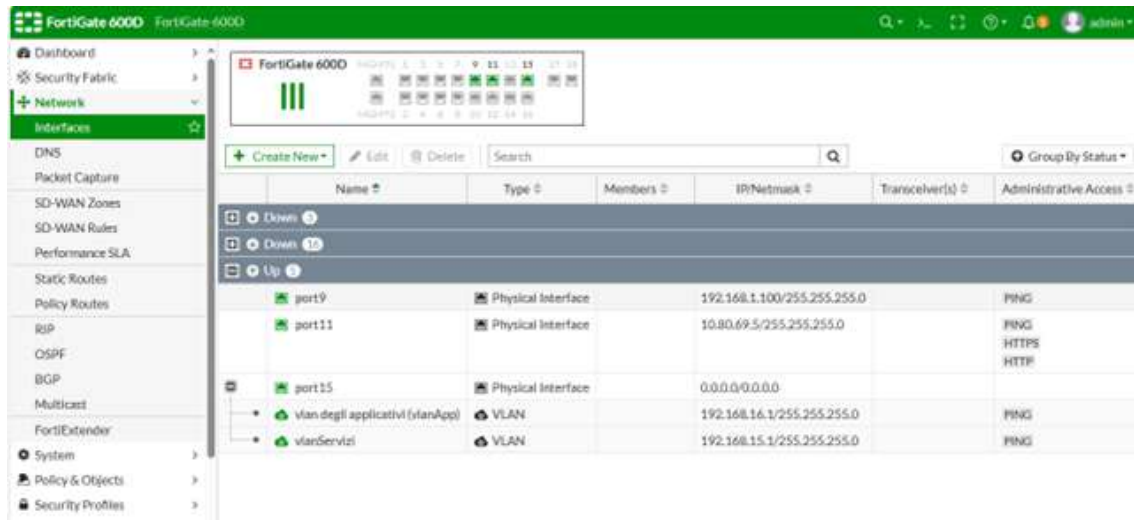
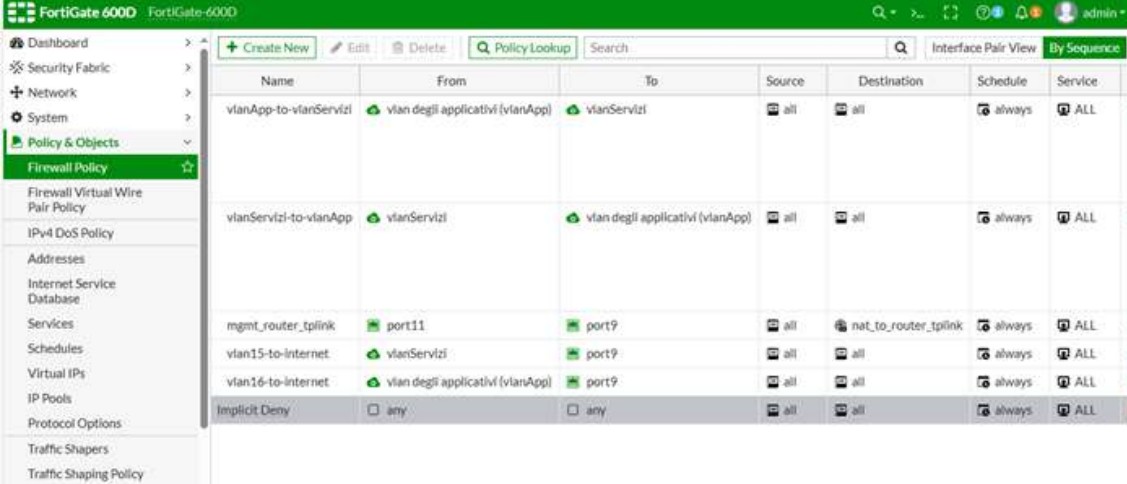


Figura 3.7: Dashboard finale delle interfacce del firewall

Successivamente, sono state configurate le policy di sicurezza. In particolare, sono state create:

- Due policy inter-VLAN: una per consentire la comunicazione dalla VLAN 16 (vlanApplicativi) verso la VLAN 15 (vlanServizi), e una seconda per il traffico nella direzione opposta;
- Due policy per l'accesso a Internet: una dalla VLAN 16 verso l'esterno e una dalla VLAN 15 verso l'esterno. Per entrambe le policy di uscita è stata abilitata la funzione di NAT (Network Address Translation), necessaria per consentire agli host interni di accedere a Internet utilizzando l'indirizzo IP pubblico del firewall, mascherando così i loro indirizzi IP privati;
- Una policy di default con azione di **deny**, utile per bloccare qualsiasi traffico non esplicitamente autorizzato.

3.3. Implementazione su dispositivi fisici



Name	From	To	Source	Destination	Schedule	Service
vlanApp-to-vlanServizi	vlan degli applicativi (vlanApp)	vlanServizi	all	all	always	ALL
vlanServizi-to-vlanApp	vlanServizi	vlan degli applicativi (vlanApp)	all	all	always	ALL
mgmt_router_tplink	port11	port9	all	nat_to_router_tplink	always	ALL
vlan15-to-internet	vlanServizi	port9	all	all	always	ALL
vlan16-to-internet	vlan degli applicativi (vlanApp)	port9	all	all	always	ALL
Implicit Deny	any	any	all	all	always	ALL

Figura 3.8: Policy di sicurezza configurate sul firewall

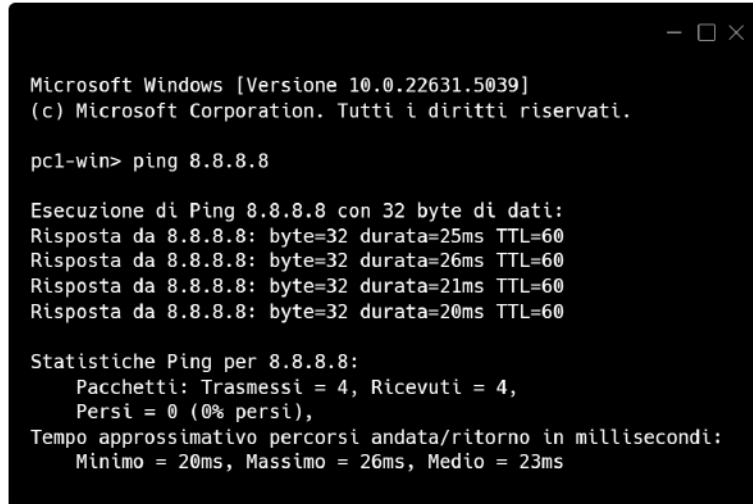
Con l'insieme delle policy sopra descritte, è stato possibile ottenere un flusso del traffico sempre controllato.

3.3.4 Router Cisco: gestione del routing

Il router Cisco è stato configurato per fornire connettività verso l'esterno, sfruttando una connessione mobile tramite una SIM inserita direttamente nel dispositivo. Questa soluzione ha permesso di abilitare l'accesso a Internet anche in assenza di una linea cablata, garantendo così la disponibilità di connettività per i test e per le comunicazioni in uscita.

3.3.5 Verifica della connettività

Per verificare l'effettivo accesso a Internet da parte degli endpoint, sono stati eseguiti comandi di `ping` verso l'indirizzo IP pubblico 8.8.8.8, ovvero uno dei server DNS messi a disposizione da Google. Questo test ha confermato che il traffico generato dai dispositivi locali veniva correttamente instradato verso l'esterno attraverso il router e il firewall, dimostrando il corretto funzionamento del collegamento alla rete Internet.



```
Microsoft Windows [Versione 10.0.22631.5039]
(c) Microsoft Corporation. Tutti i diritti riservati.

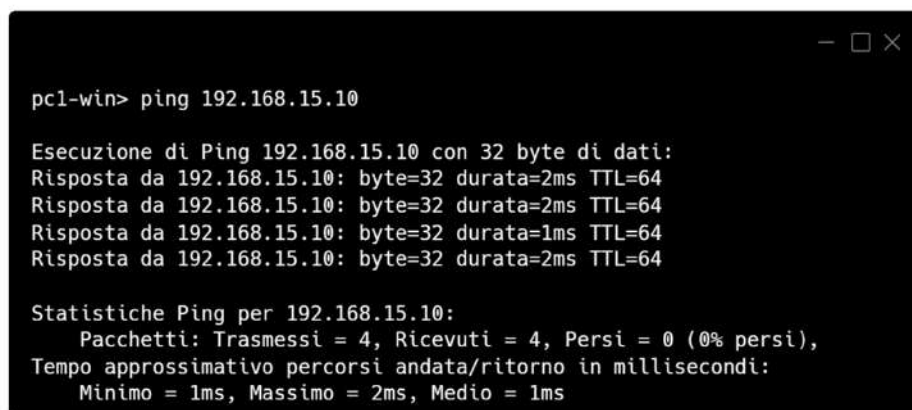
pcl-win> ping 8.8.8.8

Esecuzione di Ping 8.8.8.8 con 32 byte di dati:
Risposta da 8.8.8.8: byte=32 durata=25ms TTL=60
Risposta da 8.8.8.8: byte=32 durata=26ms TTL=60
Risposta da 8.8.8.8: byte=32 durata=21ms TTL=60
Risposta da 8.8.8.8: byte=32 durata=20ms TTL=60

Statistiche Ping per 8.8.8.8:
    Pacchetti: Trasmessi = 4, Ricevuti = 4,
    Persi = 0 (0% persi),
    Tempo approssimativo percorsi andata/ritorno in millisecondi:
        Minimo = 20ms, Massimo = 26ms, Medio = 23ms
```

Figura 3.9: Verifica della connettività Internet da un endpoint tramite comando ping

Oltre alla connettività verso l'esterno, sono state testate anche le policy di sicurezza configurate sul firewall. Per verificare la comunicazione tra VLAN, sono stati eseguiti ping tra dispositivi appartenenti a VLAN differenti. I test sono stati ripetuti in entrambe le direzioni (`vlanApplicativi` verso `vlanServizi` e viceversa) e hanno confermato che il traffico veniva correttamente autorizzato.



```
pcl-win> ping 192.168.15.10

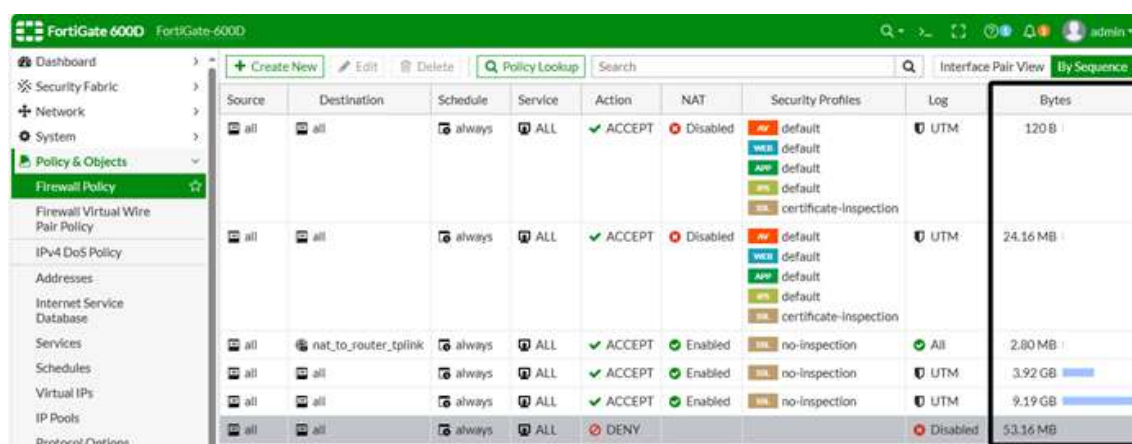
Esecuzione di Ping 192.168.15.10 con 32 byte di dati:
Risposta da 192.168.15.10: byte=32 durata=2ms TTL=64
Risposta da 192.168.15.10: byte=32 durata=2ms TTL=64
Risposta da 192.168.15.10: byte=32 durata=1ms TTL=64
Risposta da 192.168.15.10: byte=32 durata=2ms TTL=64

Statistiche Ping per 192.168.15.10:
    Pacchetti: Trasmessi = 4, Ricevuti = 4, Persi = 0 (0% persi),
    Tempo approssimativo percorsi andata/ritorno in millisecondi:
        Minimo = 1ms, Massimo = 2ms, Medio = 1ms
```

Figura 3.10: Verifica della connettività inter-VLAN

3.4. Configurazione dei servizi di rete

Per validare ulteriormente l'efficacia delle policy, durante l'esecuzione dei ping è stata osservata la dashboard delle policy del firewall FortiGate: è stato possibile verificare in tempo reale l'incremento dei contatori di traffico associati alle regole in uso, sia per le comunicazioni interne tra VLAN sia per l'accesso a Internet. Questo ha confermato che il traffico era effettivamente instradato secondo le regole configurate e che la configurazione risultava coerente e funzionante.



Source	Destination	Schedule	Service	Action	NAT	Security Profiles	Log	Bytes
all	all	always	ALL	ACCEPT	Disabled	default	UTM	120 B
all	all	always	ALL	ACCEPT	Disabled	default	UTM	24.16 MB
all	nat_to_router_tplink	always	ALL	ACCEPT	Enabled	no-inspection	All	2.80 MB
all	all	always	ALL	ACCEPT	Enabled	no-inspection	UTM	3.92 GB
all	all	always	ALL	ACCEPT	Enabled	no-inspection	UTM	9.19 GB
all	all	always	ALL	DENY			Disabled	53.16 MB

Figura 3.11: Monitoraggio del traffico delle policy firewall tramite la colonna «Bytes»

3.4 Configurazione dei servizi di rete

In seguito all'implementazione dell'infrastruttura, sono stati configurati alcuni servizi di rete fondamentali per simulare un ambiente applicativo reale all'interno della rete aziendale. In particolare, due endpoint Windows sono stati utilizzati per l'erogazione di servizi HTTP e FTP, funzionali anche alle successive fasi di monitoraggio.

3.4.1 Server HTTP

Uno degli endpoint Windows è stato configurato per offrire funzionalità di Web Server, sfruttando il ruolo integrato di Microsoft IIS (Internet Information Services). Questo componente, già incluso nel sistema operativo, è stato abilitato tramite il pannello «Attiva o disattiva funzionalità di Windows», senza la necessità di installazioni aggiuntive.

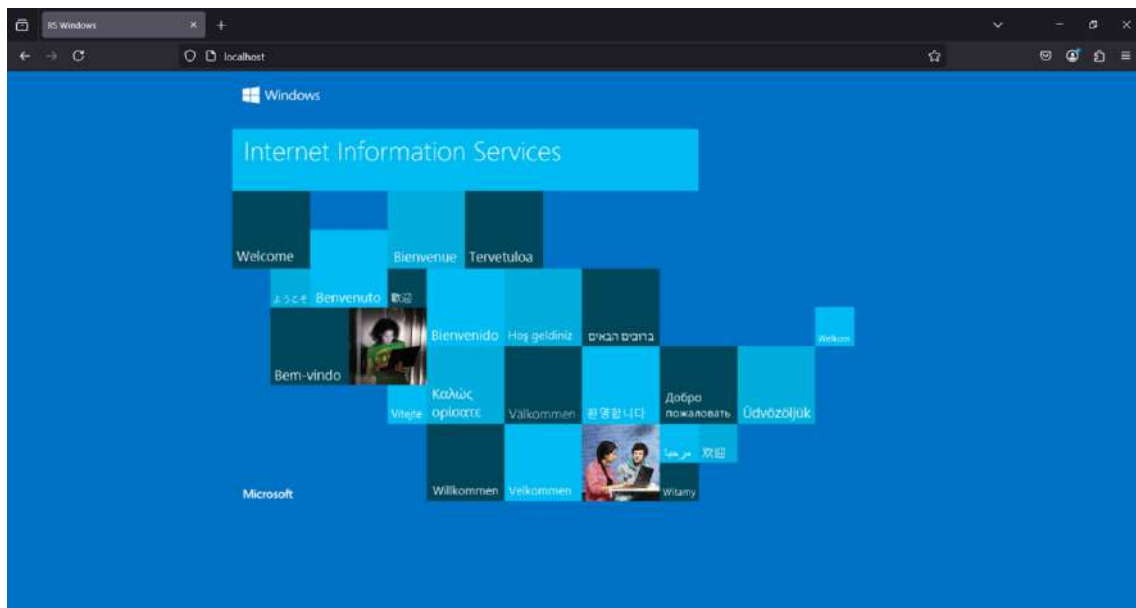


Figura 3.12: Pagina di default di IIS

Una volta attivato, IIS ha iniziato a rispondere alle richieste sulla porta 80, rendendo immediatamente disponibile una pagina HTML predefinita. Questa è stata visualizzata correttamente da un altro host della rete tramite browser web, digitando l'indirizzo IP del server nella barra degli indirizzi. La corretta visualizzazione della pagina ha confermato il buon esito della configurazione.

3.4.2 Server FTP

Il secondo endpoint Windows è stato configurato come server FTP, utilizzando il software FileZilla Server. È stata condivisa una cartella situata sul desktop contenente un file PDF di test, scelta utile per verificare il corretto funzionamento del trasferimento dati. Dopo l'installazione del programma, è stato definito un utente FTP con accesso autenticato e con permessi di lettura e scrittura sulla directory condivisa. Il server è stato impostato per accettare connessioni sulla porta 21.

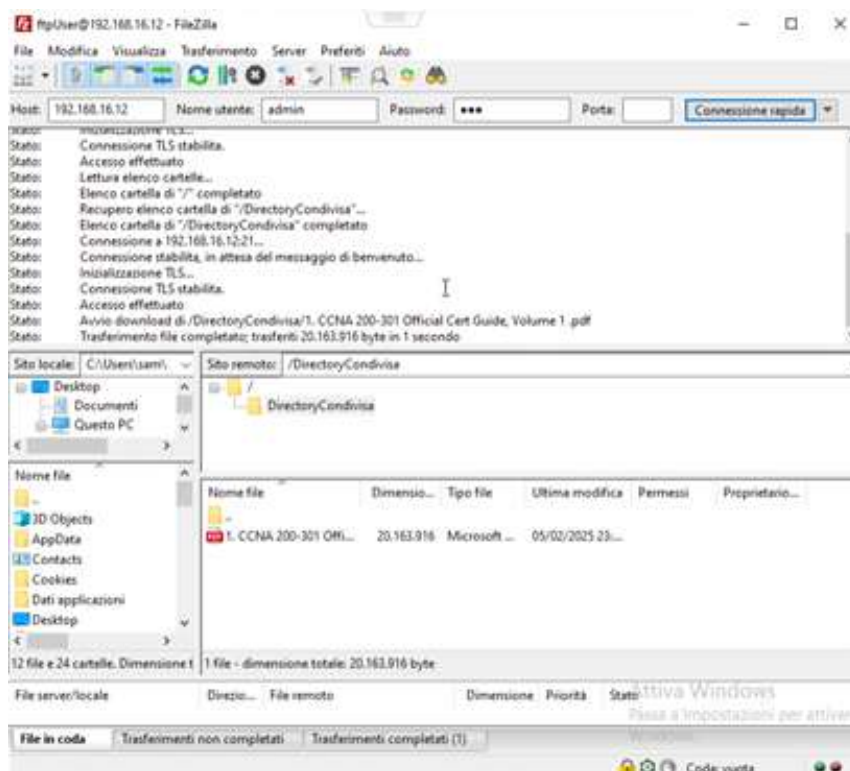


Figura 3.13: Test di connessione al server FTP da FileZilla Client

I test sono stati effettuati da un altro host nella rete, sul quale è stato installato FileZilla Client. Attraverso quest'ultimo è stata stabilita una connessione FTP al server, utilizzando l'indirizzo IP, le credenziali dell'utente configurato e accedendo alla cartella condivisa.

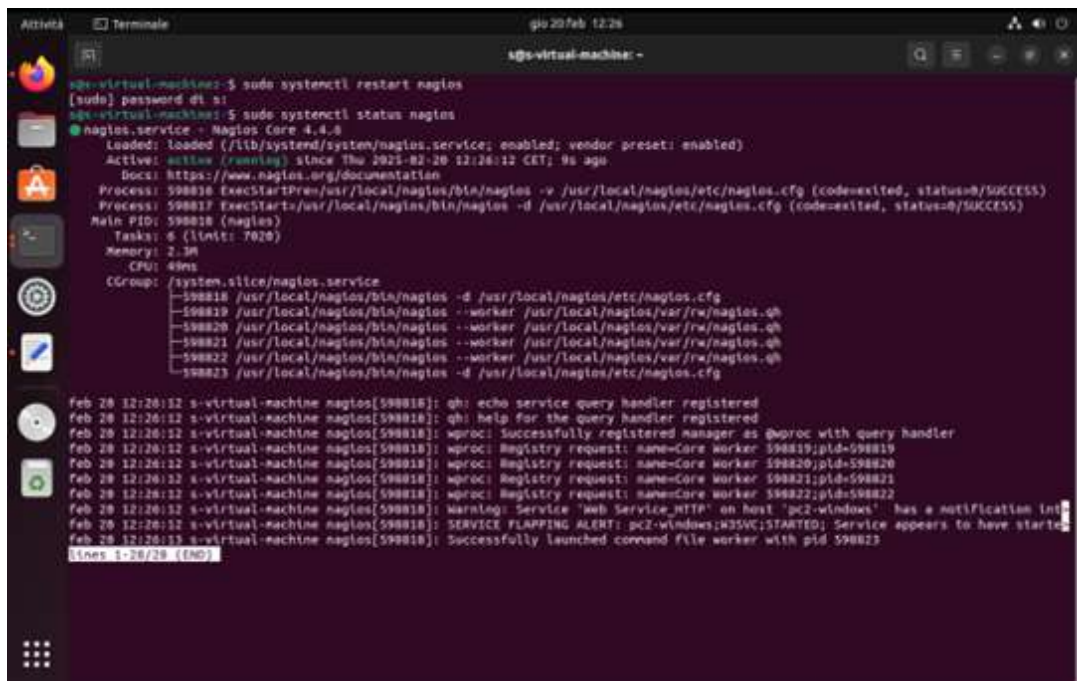
3.5 Sistema di monitoraggio

L'obiettivo principale del progetto non è stato semplicemente la creazione di una rete funzionante, ma la realizzazione di un'infrastruttura monitorata, in grado di rilevare in tempo reale eventuali malfunzionamenti e fornire visibilità sullo stato dei servizi e dei dispositivi. Per raggiungere questo scopo è stato scelto Nagios Core, una soluzione open source affidabile e ampiamente utilizzata nel settore del network monitoring.

3.5.1 Installazione di Nagios Core

Il sistema Nagios Core è stato installato sulla macchina Ubuntu, configurata in precedenza con indirizzo IP statico. Essendo un sistema basato su Linux, l'intera procedura è stata condotta da terminale, utilizzando comandi per l'installazione delle dipendenze (come Apache, PHP e le librerie necessarie), il download dei sorgenti di Nagios Core e dei plugin ufficiali per consentire l'esecuzione dei check. Inoltre, è stato creato un utente per l'autenticazione all'interfaccia web tramite `htpasswd`. Completata la configurazione, è stato effettuato un controllo di validità della configurazione mediante il comando `nagios -v` per assicurarsi che non fossero presenti errori prima del primo avvio ufficiale. Dopo aver verificato l'avvio corretto del demone Nagios e la disponibilità dell'interfaccia web, il sistema era pronto per essere popolato con gli host da monitorare.

3.5. Sistema di monitoraggio



```
s@q-virtual-machine:~$ sudo systemctl restart nagios
[sudo] password di s:
s@q-virtual-machine:~$ sudo systemctl status nagios
● nagios.service - Nagios Core 4.4.6
   Loaded: loaded (/lib/systemd/system/nagios.service; enabled; vendor preset: enabled)
   Active: active (running) since Thu 2023-02-20 12:26:12 CET; 9s ago
     Docs: https://www.nagios.org/documentation
   Process: 598816 ExecStartPre=/usr/local/nagios/bin/nagios -v /usr/local/nagios/etc/nagios.cfg (code=exited, status=0/SUCCESS)
   Process: 598817 ExecStart=/usr/local/nagios/bin/nagios -d /usr/local/nagios/etc/nagios.cfg (code=exited, status=0/SUCCESS)
   Main PID: 598818 (nagios)
     Tasks: 6 (limit: 7020)
    Memory: 2.3M
       CPU: 48ms
   CGroup: /system.slice/nagios.service
           └─ 598818 /usr/local/nagios/bin/nagios -d /usr/local/nagios/etc/nagios.cfg
              598819 /usr/local/nagios/bin/nagios --worker /usr/local/nagios/var/rw/nagios.qh
              598820 /usr/local/nagios/bin/nagios --worker /usr/local/nagios/var/rw/nagios.qh
              598821 /usr/local/nagios/bin/nagios --worker /usr/local/nagios/var/rw/nagios.qh
              598822 /usr/local/nagios/bin/nagios --worker /usr/local/nagios/var/rw/nagios.qh
              598823 /usr/local/nagios/bin/nagios -d /usr/local/nagios/etc/nagios.cfg

Feb 20 12:26:12 s-virtual-machine nagios[598818]: qh: echo service query handler registered
Feb 20 12:26:12 s-virtual-machine nagios[598818]: qh: help for the query handler registered
Feb 20 12:26:12 s-virtual-machine nagios[598818]: wproc: Successfully registered manager as @wproc with query handler
Feb 20 12:26:12 s-virtual-machine nagios[598818]: wproc: Registry request: name=Core Worker 598819;pid=598819
Feb 20 12:26:12 s-virtual-machine nagios[598818]: wproc: Registry request: name=Core Worker 598820;pid=598820
Feb 20 12:26:12 s-virtual-machine nagios[598818]: wproc: Registry request: name=Core Worker 598821;pid=598821
Feb 20 12:26:12 s-virtual-machine nagios[598818]: wproc: Registry request: name=Core Worker 598822;pid=598822
Feb 20 12:26:12 s-virtual-machine nagios[598818]: Warning: Service 'web Service_HTTP' on host 'pc2-windows' has a notification in
Feb 20 12:26:12 s-virtual-machine nagios[598818]: SERVICE FLAPPING ALERT: pc2-windows:W3SVC:STARTED: Service appears to have started
Feb 20 12:26:13 s-virtual-machine nagios[598818]: Successfully launched command file worker with pid 598823
```

Figura 3.14: Verifica dello stato del servizio Nagios

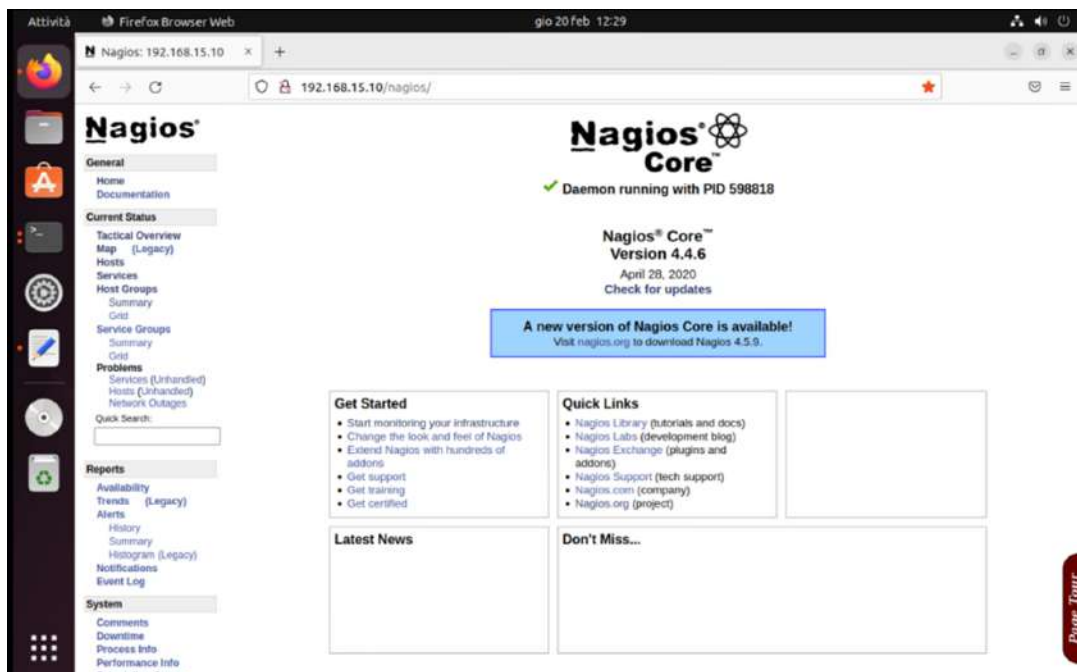


Figura 3.15: Homepage dell'interfaccia web di Nagios Core

3.5.2 Configurazione degli host con NSClient++

Per monitorare i sistemi Windows, è stato utilizzato NSClient++, un agente compatibile con Nagios. L'agente è stato installato su ciascun endpoint Windows e configurato per permettere le connessioni dal server di monitoraggio, aggiungendo l'indirizzo IP del server Ubuntu nella sezione `allowed hosts` del file `nsclient.ini`. Sono stati attivati solo i moduli necessari, tra cui `CheckSystem` e `CheckDisk`, per fornire dati relativi allo stato delle risorse (utilizzo della CPU, dello spazio su disco, ecc.).

```
# pc1windows.cfg
define host {
    use                windows-server
    host_name          pc1-windows
    alias              pc1-windows
    address            192.168.16.12
    check_period       24x7
    check_command      check-host-alive
    check_period       24x7
    notifications_enabled 1
    notification_interval 1
    notification_period 24x7
    notification_options d,r
    contacts           admin
    max_check_attempts 3
}
```

Listing 3.1: Definizione di host nel file `pc1windows.cfg`

La comunicazione tra Nagios e NSClient++ è avvenuta tramite query specifiche definite nel file `commands.cfg`, mentre i file `pc1windows.cfg` e `pc2windows.cfg` hanno descritto in dettaglio i parametri e i servizi da monitorare per ciascun host.

```
# commands.cfg

define command {
    command_name    check-host-alive
    command_line    $USER1$/check_ping -H $HOSTADDRESS$
                  -w 3000.0,80% -c 5000.0,100% -p 5
}

define command {
    command_name    check_http
    command_line    $USER1$/check_http -I $HOSTADDRESS$ $ARG1$
}

define command {
    command_name    check_ftp
    command_line    $USER1$/check_ftp -H $HOSTADDRESS$ $ARG1$
}
```

Listing 3.2: Comandi di check host, http e ftp nel file `commands.cfg`

Tra le configurazioni definite, sono stati anche impostati i valori soglia per l'invio degli alert: per esempio, l'utilizzo del disco genera un warning se lo spazio libero scende sotto il 20% e un errore critico sotto il 10%. Analogamente, l'utilizzo della CPU è monitorato con soglie personalizzate: viene generato un warning se il carico supera l'80% e un alert critico se si oltrepassa il 90%, così da ricevere segnalazioni in caso di sovraccarico. Inoltre, è stato configurato il numero massimo di tentativi di verifica consecutivi (`max_check_attempts`) prima che venga effettivamente generata una notifica: questo valore, impostato a 3, consente di evitare falsi positivi, riducendo il rischio di ricevere alert per problemi temporanei o transitori.

```
# pc1windows.cfg
define service {
    use                generic-service
    host_name          pc1-windows
    service_description CPU Load
    check_command       check_nt!CPULOAD!-l 5, 80, 90
    notifications_enabled 1
    notification_options w,c,r
    contacts            admin
    max_check_attempts 3
    flap_detection_enabled 0
}

define service {
    use                generic-service
    host_name          pc1-windows
    service_description Memory Usage
    check_command       check_nt!MEMUSE!-w 80 -c 90
    notifications_enabled 1
    max_check_attempts 3
}
```

Listing 3.3: Definizione di soglie e tentativi per i controlli su host e servizi

Dopo aver riavviato il servizio Nagios, è stato possibile visualizzare lo stato dei dispositivi tramite l'interfaccia web. Quest'ultima fornisce informazioni dettagliate come il tempo trascorso dall'ultimo check eseguito e la durata dello stato corrente (in caso di errore, warning o stato OK), offrendo una panoramica precisa e continua sull'affidabilità dell'infrastruttura monitorata.

3.6. Sistema di notifiche e alert automatici

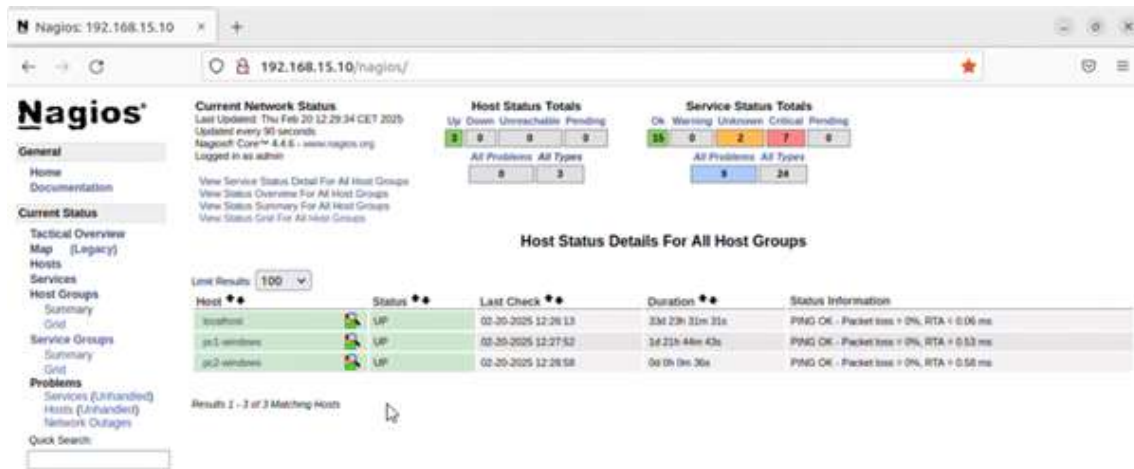


Figura 3.16: Interfaccia web di Nagios dello stato degli host

3.6 Sistema di notifiche e alert automatici

Per completare il sistema di monitoraggio, è stato configurato un meccanismo di notifiche automatiche via email, in modo da avvisare tempestivamente gli amministratori di rete in caso di anomalie o interruzioni dei servizi.

3.6.1 Installazione e configurazione di Postfix

Per l'invio delle email è stato utilizzato Postfix, un mail transfer agent installato sulla macchina Ubuntu. La configurazione è stata eseguita in modalità «send-only», sufficiente per inoltrare notifiche generate da Nagios verso un indirizzo email predefinito. La configurazione è avvenuta selezionando la modalità «Internet Site» durante il processo guidato di installazione del pacchetto. A seguito di questa scelta, è stato definito il dominio locale del sistema, che rappresenta l'identità del mittente nelle comunicazioni email. Sono stati modificati i file di configurazione principali, in particolare `main.cf`, dove sono stati indicati il nome dell'host, il dominio e il `relayhost` utilizzato per l'invio dei messaggi tramite un server SMTP esterno.

3.6. Sistema di notifiche e alert automatici

È stato configurato l'inoltro attraverso un account Gmail di prova, per simulare la casella di posta dell'amministratore di rete, specificando l'SMTP server di Google e le relative credenziali di autenticazione per permettere l'invio dei messaggi.

```
# commands.cfg
define command {
    command_name    notify-host-by-email
    command_line    /usr/bin/printf
                    "%b" "***** Nagios HOST *****\n\n
                    Notification Type: $NOTIFICATIONTYPE$\n
                    Host: $HOSTNAME$\n State: $HOSTSTATE$\n
                    Address: $HOSTADDRESS$\n Info: $HOSTOUTPUT$\n
                    Date/Time: $LONGDATETIME$\n"
                    |mail -s "** $NOTIFICATIONTYPE$
                    Host Alert: $HOSTNAME$ is $HOSTSTATE$ **"
                    $CONTACTEMAIL$
}

define command {
    command_name    notify-service-by-email
    command_line    /usr/bin/printf
                    "%b" "***** Nagios SERVIZIO *****\n\n
                    Notification Type: $NOTIFICATIONTYPE$\n
                    Service: $SERVICEDESC$\n Host: $HOSTALIAS$\n
                    Address: $HOSTADDRESS$\n
                    State: $SERVICESTATE$\n
                    Date/Time: $LONGDATETIME$\n
                    Additional Info:\n\n$SERVICEOUTPUT$\n"
                    |mail -s "** $NOTIFICATIONTYPE$
                    Service Alert: $HOSTALIAS$/$SERVICEDESC$
                    is $SERVICESTATE$ **" $CONTACTEMAIL$
}
```

Listing 3.4: Comandi per la generazione delle email per problemi sugli host e servizi

3.6.2 Test e gestione degli alert via email

Per verificare l'efficacia del sistema di notifica, sono stati eseguiti diversi test emulando condizioni di errore reali. Tra questi, l'arresto temporaneo dell'host `pc1-windows` ha causato il passaggio dello stato da "UP" a "DOWN", come visibile nell'interfaccia web di Nagios. Questo ha generato automaticamente una notifica email indirizzata all'amministratore di rete, con oggetto "Host Alert" e contenente informazioni dettagliate sul tipo di errore, l'indirizzo IP e il momento esatto del rilevamento. Una volta ripristinata la connettività dell'host, il sistema ha eseguito nuovamente i check, determinando il ritorno allo stato "UP". Anche in questo caso è stata inviata una notifica via email che ha confermato il recupero della situazione e il ripristino dell'operatività dell'host monitorato.

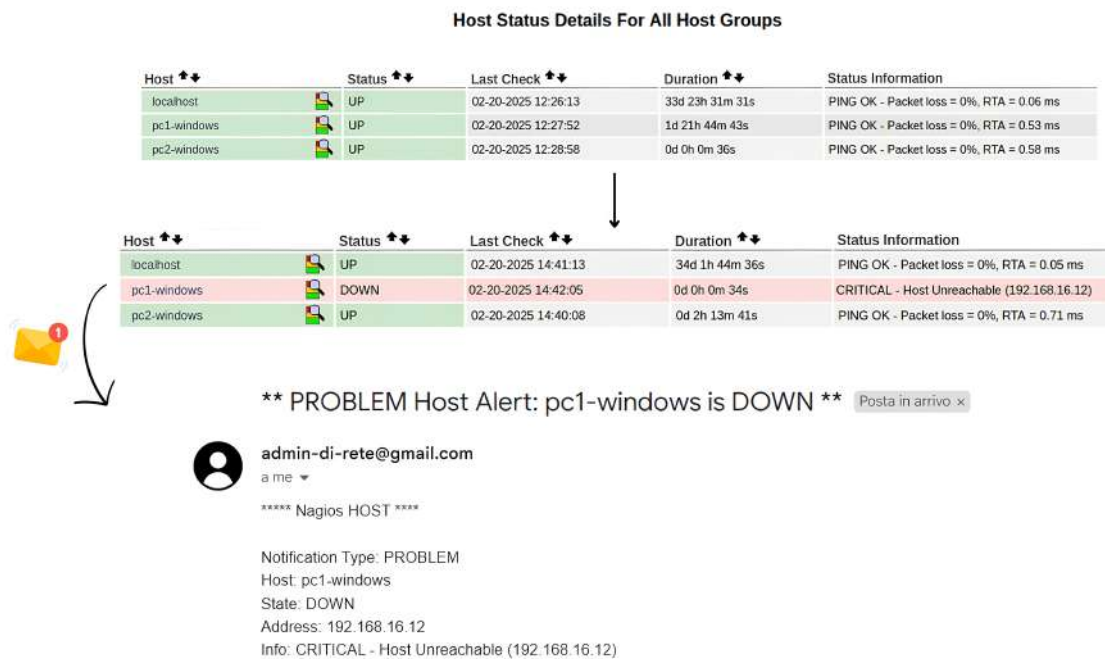


Figura 3.17: Notifica email per stato DOWN di un host

È stato inoltre verificato il comportamento del sistema di notifica in caso di indisponibilità dei servizi applicativi configurati sugli host monitorati, nello specifico HTTP e FTP. Per simulare un malfunzionamento, sono stati disattivati temporaneamente

3.6. Sistema di notifiche e alert automatici

i servizi Web (IIS) e FTP (FileZilla Server) sulle rispettive macchine Windows. Il sistema di monitoraggio ha rilevato l'inaccessibilità delle porte 80 e 21, attivando così le notifiche via email. Anche in questo caso, le email ricevute indicavano chiaramente il tipo di servizio non raggiungibile, l'indirizzo IP dell'host e lo stato di errore.

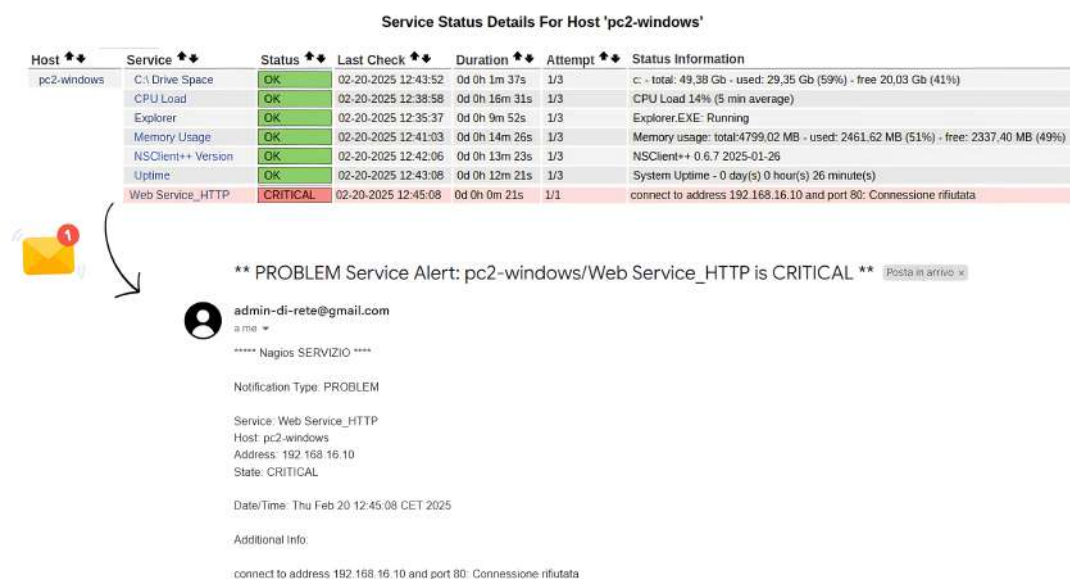


Figura 3.18: Notifica email generata per inattività del servizio HTTP sull'host pc2windows

Analogamente, sono stati simulati casi di sovraccarico della CPU e saturazione dello spazio su disco, superando le soglie definite per warning e critical: anche in questi casi sono state ricevute email puntuali con dettagli relativi allo stato, all'orario del rilevamento e alla descrizione del problema. Oltre alle segnalazioni per condizioni di criticità, il sistema invia automaticamente una notifica di stato "OK" una volta che i valori monitorati rientrano nei limiti prestabiliti. Questo comportamento garantisce all'amministratore visibilità costante sul ritorno alla normalità delle condizioni operative, confermando che il problema è stato risolto.

3.6. Sistema di notifiche e alert automatici



Figura 3.19: Notifica email generata per superamento della soglia di WARNING del carico CPU

Le notifiche sono state osservate su una casella Gmail configurata a tale scopo, dimostrando l'integrazione funzionante tra Nagios, Postfix e il server SMTP esterno. Complessivamente, i test hanno dimostrato l'affidabilità del sistema di notifiche configurato: ogni scenario di malfunzionamento ha generato una notifica puntuale e dettagliata, mentre gli stati di normalità sono stati confermati tramite messaggi di ritorno allo stato "OK"/"UP".

Capitolo 4

Conclusioni

Il progetto di tesi ha avuto come obiettivo la progettazione, realizzazione e monitoraggio di un'infrastruttura di rete, affrontando in modo completo tutte le fasi necessarie: dalla definizione dell'architettura fino alla configurazione di un sistema di monitoraggio efficiente e automatizzato. Il lavoro ha permesso di verificare sul campo l'efficacia delle soluzioni adottate, dimostrando come un sistema di monitoraggio ben configurato sia in grado di rilevare tempestivamente malfunzionamenti, criticità e variazioni nello stato dei servizi e degli host. I test effettuati e le notifiche ricevute hanno confermato la correttezza dell'approccio progettuale e la funzionalità dell'intera infrastruttura. L'esperienza maturata nel corso dello sviluppo mi ha permesso di comprendere in modo più profondo e concreto quanto appreso durante il percorso universitario. Più che semplicemente applicare le conoscenze teoriche, questo progetto ha rappresentato l'occasione per trasformarle in competenze pratiche, rafforzandone il significato attraverso la realizzazione di un caso reale. In conclusione, il lavoro svolto può ritenersi pienamente riuscito, avendo raggiunto gli obiettivi previsti e portato alla realizzazione di un'infrastruttura funzionale, stabile e pronta per contesti operativi concreti.

Bibliografia

- [1] Cisco. *Che cos'è un firewall e come funziona?* [Online]. URL: https://www.cisco.com/c/it_it/products/what-is-a-firewall.html.
- [2] Connect S.p.A. *A cosa serve segmentare una rete?* [Online]. URL: <https://connectspa.it/news/a-cosa-serve-segmentare-una-rete/>.
- [3] FileZilla Project. *FileZilla Server*. [Online]. URL: <https://filezilla-project.org/>.
- [4] Fortinet. *Firewall Policy Configuration Guide*. [Online]. URL: <https://docs.fortinet.com>.
- [5] Fortinet. *FortiGate 200F Product Data Sheet*. [Online]. URL: <https://www.fortinet.com>.
- [6] Fortinet. *Network Monitoring*. [Online]. URL: <https://www.fortinet.com/it/resources/cyberglossary/network-monitoring>.
- [7] GlobalYo. *The importance of network monitoring: Ensuring optimal performance and security*. [Online]. 2019. URL: <https://www.globalyo.com/it/blog/the-importance-of-network-monitoring-ensuring-optimal-performance-and-security/>.
- [8] GNS3. *Graphical Network Simulator*. [Online]. URL: <https://www.gns3.com/>.
- [9] IBM. *Cos'è un firewall?* [Online]. URL: <https://www.ibm.com/it-it/topics/firewall>.

- [10] ManageEngine. *Basics of network monitoring*. [Online]. URL: <https://www.manageengine.com/network-monitoring/basics-of-network-monitoring.html>.
- [11] MobaTek. *MobaXterm terminal for Windows*. [Online]. URL: <https://mobaxterm.mobatek.net/>.
- [12] Nagios Support. *NRPE - Nagios Remote Plugin Executor*. [Online]. URL: <https://support.nagios.com/kb/print-141.html>.
- [13] Nagios.org. *About Nagios Core*. [Online]. URL: <https://www.nagios.org/projects/nagios-core/>.
- [14] NSClient++. *NSClient++ Documentation*. [Online]. URL: <https://docs.nscient.org/>.
- [15] Postfix. *Postfix Documentation*. [Online]. URL: <http://www.postfix.org/documentation.html>.
- [16] Red Hat. *Infrastruttura di rete: cos'è, come funziona e perché è importante*. [Online]. URL: <https://www.redhat.com/it/partners/network-infrastructure>.
- [17] Red Hot Cyber. *Wireshark: Uno squalo per amico*. [Online]. URL: <https://www.redhotcyber.com/post/wireshark-uno-squalo-per-amico/>.
- [18] VMware. *Virtual LANs (VLANs)*. [Online]. URL: <https://docs.vmware.com>.